@WeRateDogs

Wrangling

Goal of this part of the project

Wrangle WeRateDogs Twitter data to create analyses and visualizations. WeRateDogs is a twitter page designated to rating dogs on their appearences and stories. It runs since 2015 under @dog_rates name. First of all the data must be wrangled. To do so, data must be gathered, assesed and cleaned.

Gathering data

Data Sources

1. *archive_enhanced* This file contains information about tweets and is provided by Udacity to all students. It can be imported directly via *pd.read_csv.*

2. *dog_predictions* It consists in a table full of image predictions (the top three only) alongside each tweet ID, image URL, and the image number that corresponded to the most confident prediction about what breed of dog (or other objects, animal, etc.) is present in each tweet according to a neural network. The file is hosted on

Udacity's servers and downloaded programmatically using the requests library and the provided url.

3. *json_twitter* The process of obtaining data is made from creating an API object with Tweepy on Twitter. Download the JSON of each tweet in a .txt file so that each tweet occupies a line.

Data Assessing
Quality and Tidiness issues

According to what the course states, Assessing Data have to do with quality and tidiness.

Quality issues pertain to content. Low quality data is also known as dirty data. There are four dimensions of quality data:

1. **Completeness:** do we have all of the records that we should? Do we have missing records or not? Are there specific rows, columns, or cells missing?
2. **Validity:** we have the records, but they're not valid, i.e., they don't conform to a defined schema. A schema is a defined set of rules for data. These rules can be real-world constraints (e.g. negative height is impossible) and table-specific constraints (e.g. unique key constraints in tables).
3. **Accuracy:** inaccurate data is wrong data that is valid. It adheres to the defined schema, but it is still incorrect. Example: a patient's weight that is 5 lbs too heavy because the scale was faulty.
4. **Consistency:** inconsistent data is both valid and accurate, but there are multiple correct ways of referring to the same thing. Consistency, i.e., a standard format, in columns that represent the same data across tables and/or within tables is desired.

Tidiness issues pertain to structure. These structural problems generally prevent easy analysis. Untidy data is also known as messy data. The requirements for tidy data are:

1. Each variable forms a column.

2. Each observation forms a row.
3. Each type of observational unit forms a table.

And according to this structure:

Quality Issues	Completeness	Validity	Accuracy	Consistency
archive_enhanced (df)	There are 2297 expanded_urls non-null object, so there are missing values. There are missing dog names (they appear as None.	There are denominators different to 10 Numerators can be extremely away from 10	Observation let see that some names are not rally names. **a, an** are good examples.	Retweeted_status_timestamp and timestamp appear as object instead of datetimes
dog_predictions (df)	missing data (only has 2075 entries instead of 2356)		p1, p2, p3 have some accuarcy spelling mistakes. Not always first letter is capital/lower.	
json_twitter (df)	missing data (only has 2339 entries instead of 2356).			
twitter_archive_master_df (df)	Source twitter_archive_master_df shorting			

Tidiness Issues	
archive_enhanced (df)	There are multiple columns containing the same variable. Dog Type appear in three columns doggo, floofer, pupper and puppo.

Finally, a common tidines issue is that the dataframes archive_enhanced and json_twitter contain information about tweets, so they ara the same observation and they would be in a

seme dataframe and mergin the three of them in a single dataframe named twitter_archive_master_df.

Data Cleaning

Quality and Tidiness issues

The detailed information issues to be fixed are cleaned in the following order.

1. *Missing values in expanded_urls*
   There are 2297 expanded_urls non-null object, so there are missing values. Tweet_id is used to add the number at the end of the url beggining pattern https://twitter.com/dog_rates/status/.... (5.1)

2. *Remove any tweet ids in the archive table that aren't in the predictions table*
   As there is missing data (only has 2339 entries instead of 2356) and , as well, missing data (only has 2075 entries instead of 2356) in dataframes dog_predictions and json_twitter. The intersecction of them is kept in form of the minimum subset. (5.2)

3. *Merge the 2 dataframes corresponding to tweet information to the same dataframe*
   There is no more than pd.merge direct application to fix this tidiness issue.(5.3)

4. *There are missing dog names (they appear as None)*
   We are going to change the names of dogs such as a, an , the , O , None....visually or programatically discovered.(5.4)

5. *There are denominators different to 10*
   Remove them (they are 13).(5.5)

6. *Numerators can be extremely away from 10*
   Some of the dog ratings are exaggerated, 1776/10 for instance !!! So to avood a distorsion, better drop the large values . What a normal rate would/should be? Is reasonable betwwen 10/10 and 20/10 but here is not a good idea cause there are good a fourth of the values under 10/10. So , better take into consideration values with a number of usages which represent at least a 0.5% of the total. It means 0.5% of 2048~ 10,25. So ,let's consider only 10 or more.(5.6)

7. *Convert timestamp and retweeted_status_timestamp to data type datetime*

   Direct application pd.to_datetime (5.7)

8. *p1, p2, p3 have some accuarcy spelling mistakes. Not always first letter is capital/lower*

   Convert all the p1 p2 and p3 to lowercase (5.8)

9. *There are multiple columns containing the same variable. Dog Type appear in three columns doggo, floofer, pupper and puppo*

   A new variable – 'internet_dog_terminology' is created to show the four dog stages, drop the four columns, and fill the empty with NaN. (5.9)

10. *Merge the two df in one called twitter_archive_master_df*

    Merging the two: tacf and dpc_df in one. (5.10)

11. *Source twitter_archive_master_df shorting*

    Cut the unusefull part from source srting. (5.11)

Data Saving

Converting data and saving

The conversion of final df to csv using pandas runs afterthe following input:

```
twitter_archive_master_df.to_csv('twitter_archive_master.csv', encoding='utf-8')
```

Eudald Escribà

6 April 2019